### 정리하기

# 1.데이터 준비, 패키지 준비  
mpg <- as.data.frame(ggplot2::mpg) # 데이터 불러오기

midwest <- as.data.frame(ggplot2::midwest) # 데이터 불러오기  
library(dplyr) # dplyr 로드  
library(ggplot2) # ggplot2 로드  
  
# 2.데이터 파악  
head(mpg) # Raw 데이터 앞부분  
tail(mpg) # Raw 데이터 뒷부분  
View(mpg) # Raw 데이터 뷰어창에서 확인  
dim(mpg) # 차원  
str(mpg) # 속성  
summary(mpg) # 요약 통계량

# 3.변수명 수정  
mpg <- rename(mpg, company = manufacturer)  
  
# 4.파생변수 생성  
mpg$total <- (mpg$cty + mpg$hwy)/2 # 변수 조합  
mpg$test <- ifelse(mpg$total >= 20, "pass", "fail") # 조건문 활용  
  
# 5.빈도 확인  
table(mpg$test) # 빈도표 출력  
qplot(mpg$test) # 막대 그래프 생성

## 분석 도전

ggplot2 패키지에는 미국 동북중부 437개 지역의 인구통계 정보를 담은 midwest라는 데이터가 포함되어 있습니다. midwest 데이터를 사용해 데이터 분석 문제를 해결해보세요.

* 문제1. ggplot2의 midwest 데이터를 데이터 프레임 형태로 불러와서 데이터의 특성을 파악하세요.
* 문제2. poptotal(전체 인구)을 total로, popasian(아시아 인구)을 asian으로 변수명을 수정하세요.
* 문제3. total, asian 변수를 이용해 '전체 인구 대비 아시아 인구 백분율' 파생변수를 만들고, 히스토그램을 만들어 도시들이 어떻게 분포하는지 살펴보세요.
* 문제4. 아시아 인구 백분율 전체 평균을 구하고, 평균을 초과하면 "large", 그 외에는 "small"을 부여하는 파생변수를 만들어 보세요.
* 문제5. "large"와 "small"에 해당하는 지역이 얼마나 되는지, 빈도표를 만들어 확인해 보세요.

**[자유자재로 데이터 가공하기]**



[원하는 형태로 데이터 가공하기]

#### 데이터 전처리(Preprocessing) - dplyr 패키지

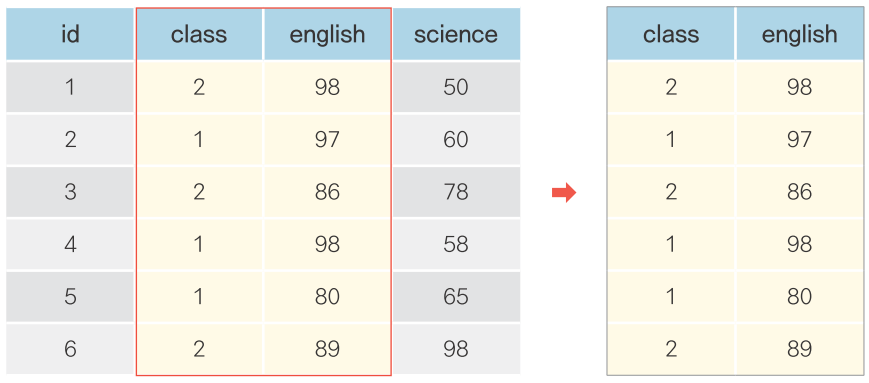
|  |  |
| --- | --- |
| 함수 | 기능 |
| **filter()** | **행 추출** |
| **select()** | **열(변수) 추출** |
| **arrange()** | **정렬** |
| **mutate()** | **변수 추가(새필드)** |
| **summarise()** | **통계치 산출** |
| **group\_by()** | **집단별로 나누기** |
| **left\_join()** | **데이터 합치기(열)** |
| **bind\_rows()** | **데이터 합치기(행)** |

### 혼자서 해보기

mpg 데이터를 이용해 분석 문제를 해결해 보세요.

* Q1. 자동차 배기량에 따라 고속도로 연비가 다른지 알아보려고 합니다. displ(배기량)이 4 이하인 자동차와 5 이상인 자동차 중 어떤 자동차의 hwy(고속도로 연비)가 평균적으로 더 높은지 알아보세요.
* Q2. 자동차 제조 회사에 따라 도시 연비가 다른지 알아보려고 합니다. "audi"와 "toyota" 중 어느 manufacturer(자동차 제조 회사)의 cty(도시 연비)가 평균적으로 더 높은지 알아보세요.
* Q3. "chevrolet", "ford", "honda" 자동차의 고속도로 연비 평균을 알아보려고 합니다. 이 회사들의 자동차를 추출한 뒤 hwy 전체 평균을 구해보세요.

**[필요한 변수만 추출하기]**

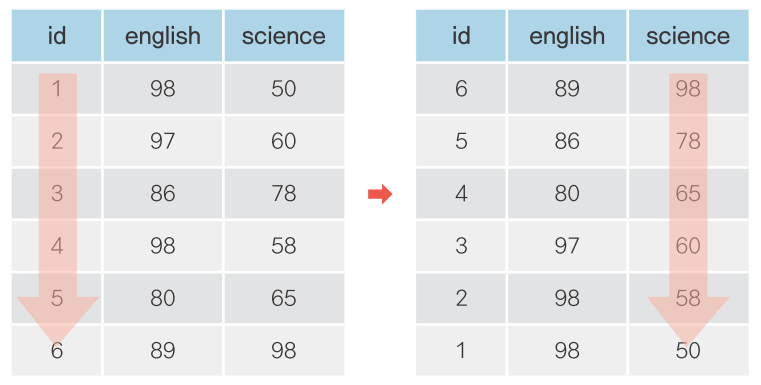


### 혼자서 해보기

mpg 데이터를 이용해서 분석 문제를 해결해보세요.

* Q1. mpg 데이터는 11개 변수로 구성되어 있습니다. 이 중 일부만 추출해서 분석에 활용하려고 합니다. mpg 데이터에서 class(자동차 종류), cty(도시 연비) 변수를 추출해 새로운 데이터를 만드세요. 새로 만든 데이터의 일부를 출력해서 두 변수로만 구성되어 있는지 확인하세요.
* Q2. 자동차 종류에 따라 도시 연비가 다른지 알아보려고 합니다. 앞에서 추출한 데이터를 이용해서 class(자동차 종류)가 "suv"인 자동차와 "compact"인 자동차 중 어떤 자동차의 cty(도시 연비)가 더 높은지 알아보세요.

**[순서대로 정렬하기]**

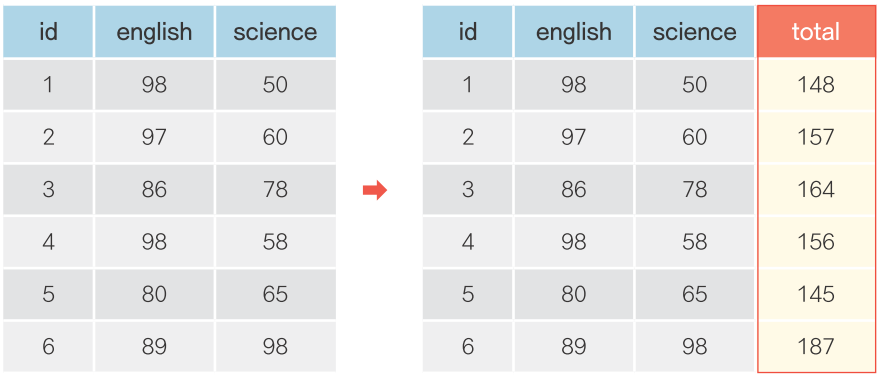


### 혼자서 해보기

mpg 데이터를 이용해서 분석 문제를 해결해보세요.

* "audi"에서 생산한 자동차 중에 어떤 자동차 모델의 hwy(고속도로 연비)가 높은지 알아보려고 합니다. "audi"에서 생산한 자동차 중 hwy가 1~5위에 해당하는 자동차의 데이터를 출력하세요.

**[파생변수 추가하기]**



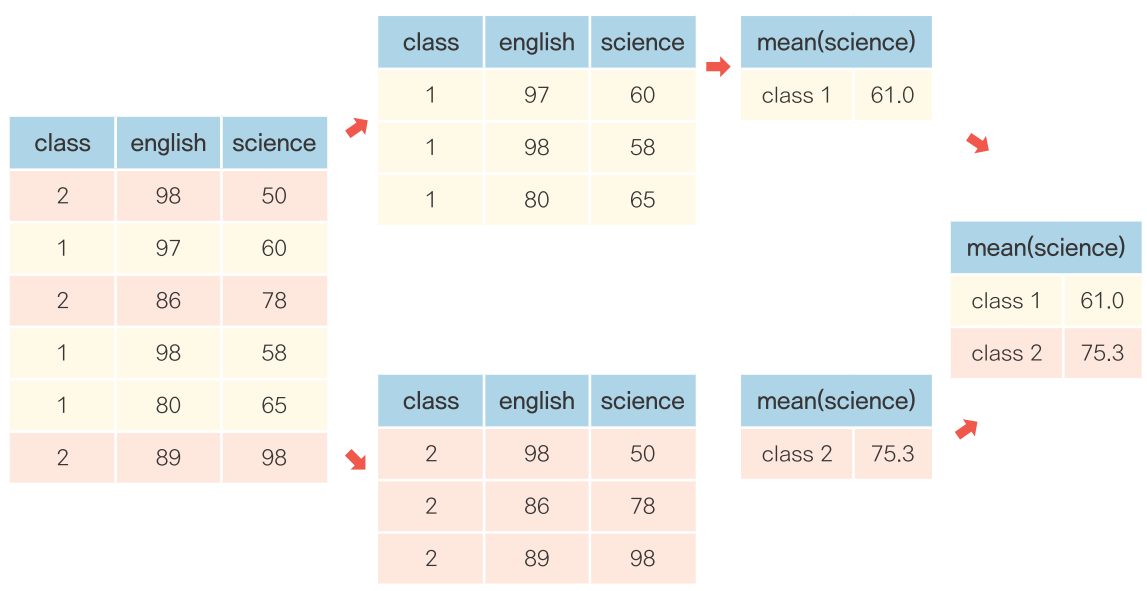
### 혼자서 해보기

mpg 데이터를 이용해서 분석 문제를 해결해보세요.

mpg 데이터는 연비를 나타내는 변수가 hwy(고속도로 연비), cty(도시 연비) 두 종류로 분리되어 있습니다. 두 변수를 각각 활용하는 대신 하나의 통합 연비 변수를 만들어 분석하려고 합니다.

* Q1. mpg 데이터 복사본을 만들고, cty와 hwy를 더한 '합산 연비 변수'를 추가하세요.
* Q2. 앞에서 만든 '합산 연비 변수'를 2로 나눠 '평균 연비 변수'를 추가세요.
* Q3. '평균 연비 변수'가 가장 높은 자동차 3종의 데이터를 출력하세요.
* Q4. 1~3번 문제를 해결할 수 있는 하나로 연결된 dplyr 구문을 만들어 출력하세요. 데이터는 복사본 대신 mpg 원본을 이용하세요.

**[집단별로 요약하기]**



#### 자주 사용하는 요약통계량 함수

|  |  |
| --- | --- |
| * 함수 | * 의미 |
| * mean() | * 평균 |
| * sd() | * 표준편차 |
| * sum() | * 합계 |
| * median() | * 중앙값 |
| * min() | * 최솟값 |
| * max() | * 최댓값 |
| * n() | * 빈도 |

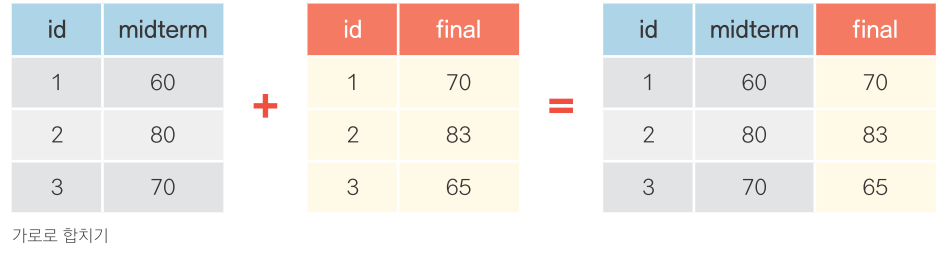
### 혼자서 해보기

mpg 데이터를 이용해서 분석 문제를 해결해 보세요.

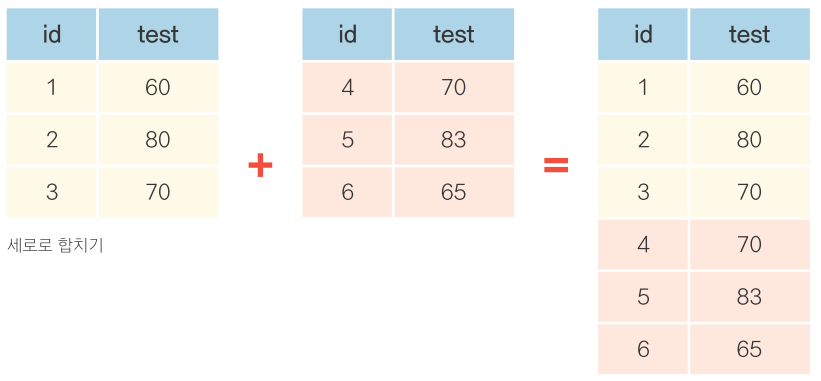
* Q1. mpg 데이터의 class는 "suv", "compact" 등 자동차를 특징에 따라 일곱 종류로 분류한 변수입니다. 어떤 차종의 연비가 높은지 비교해보려고 합니다. class별 cty 평균을 구해보세요.
* Q2. 앞 문제의 출력 결과는 class 값 알파벳 순으로 정렬되어 있습니다. 어떤 차종의 도시 연비가 높은지 쉽게 알아볼 수 있도록 cty 평균이 높은 순으로 정렬해 출력하세요.
* Q3. 어떤 회사 자동차의 hwy(고속도로 연비)가 가장 높은지 알아보려고 합니다. hwy 평균이 가장 높은 회사 세 곳을 출력하세요.
* Q4. 어떤 회사에서 "compact"(경차) 차종을 가장 많이 생산하는지 알아보려고 합니다. 각 회사별 "compact" 차종 수를 내림차순으로 정렬해 출력하세요.

**[데이터 합치기]**

#### 가로로 합치기



#### 세로로 합치기



### 혼자서 해보기

mpg 데이터를 이용해서 분석 문제를 해결해 보세요.

mpg 데이터의 fl 변수는 자동차에 사용하는 연료(fuel)를 의미합니다. 아래는 자동차 연료별 가격을 나타낸 표입니다.

|  |  |  |
| --- | --- | --- |
| * fl | * 연료 종류 | * 가격(갤런당 USD) |
| * c | * CNG | * 2.35 |
| * d | * diesel | * 2.38 |
| * e | * ethanol E85 | * 2.11 |
| * p | * premium | * 2.76 |
| * r | * regular | * 2.22 |

우선 이 정보를 이용해서 연료와 가격으로 구성된 데이터 프레임을 만들어 보세요.

fuel <- data.frame(fl = c("c", "d", "e", "p", "r"),  
 price\_fl = c(2.35, 2.38, 2.11, 2.76, 2.22),  
 stringsAsFactors = F)  
fuel # 출력

## fl price\_fl  
## 1 c 2.35  
## 2 d 2.38  
## 3 e 2.11  
## 4 p 2.76  
## 5 r 2.22

* Q1. mpg 데이터에는 연료 종류를 나타낸 fl 변수는 있지만 연료 가격을 나타낸 변수는 없습니다. 위에서 만든 fuel 데이터를 이용해서 mpg 데이터에 price\_fl(연료 가격) 변수를 추가하세요.
* Q2. 연료 가격 변수가 잘 추가됐는지 확인하기 위해서 model, fl, price\_fl 변수를 추출해 앞부분 5행을 출력해 보세요.

#### 힌트

Q1. left\_join()을 이용해서 mpg 데이터에 fuel 데이터를 합치면 됩니다. 두 데이터에 공통으로 들어있는 변수를 기준으로 삼아야 합니다.

Q2. select()와 head()를 조합하면 됩니다.

### 분석 도전

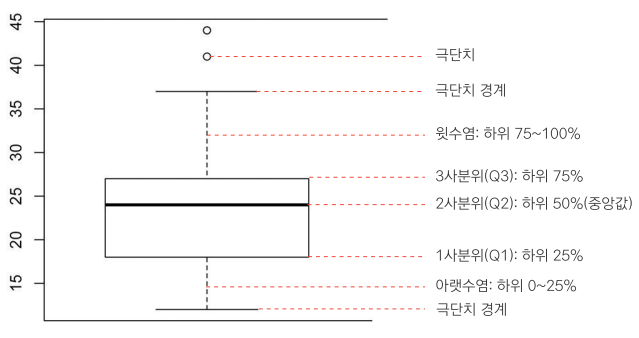
미국 동북중부 437개 지역의 인구통계 정보를 담고 있는 midwest 데이터를 사용해 데이터 분석 문제를 해결해 보세요. midwest는 ggplot2 패키지에 들어 있습니다.

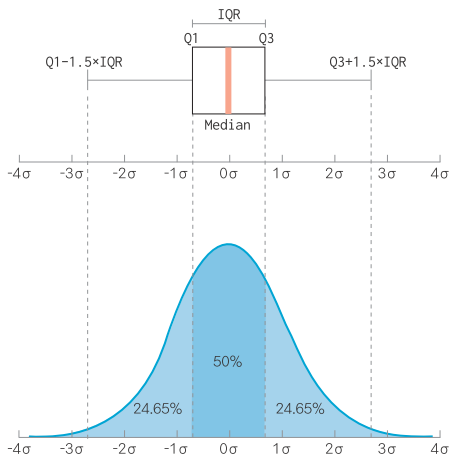
* 문제1. popadults는 해당 지역의 성인 인구, poptotal은 전체 인구를 나타냅니다. midwest 데이터에 '전체 인구 대비 미성년 인구 백분율' 변수를 추가하세요.
* 문제2. 미성년 인구 백분율이 가장 높은 상위 5개 county(지역)의 미성년 인구 백분율을 출력하세요.
* 문제3. 분류표의 기준에 따라 미성년 비율 등급 변수를 추가하고, 각 등급에 몇 개의 지역이 있는지 알아보세요.

|  |  |
| --- | --- |
| * 분류 | * 기준 |
| * large | * 40% 이상 |
| * middle | * 30% ~ 40% 미만 |
| * small | * 30% 미만 |

* 문제4. popasian은 해당 지역의 아시아인 인구를 나타냅니다. '전체 인구 대비 아시아인 인구 백분율' 변수를 추가하고, 하위 10개 지역의 state(주), county(지역명), 아시아인 인구 백분율을 출력하세요.
* 상자그림으로 극단치 기준 정해서 제거하기
* mpg <- as.data.frame(ggplot2::mpg)  
  boxplot(mpg$hwy)
* 정상범위 기준 정해서 벗어나면 결측 처리

|  |  |
| --- | --- |
| * 판단 기준 | * 예 |
| * 논리적 판단 | * 성인 몸무게 40kg~150kg 벗어나면 극단치 |
| * 통계적 판단 | * 상하위 0.3% 극단치 또는 상자그림 1.5 IQR 벗어나면 극단치 |

* 



### 혼자서 해보기

mpg 데이터를 이용해서 분석 문제를 해결해 보세요.

우선 mpg 데이터를 불러와서 일부러 이상치를 만들겠습니다. drv(구동방식) 변수의 값은 4(사륜구동), f(전륜구동), r(후륜구동) 세 종류로 되어있습니다. 몇 개의 행에 존재할 수 없는 값 k를 할당하겠습니다. cty(도시 연비) 변수도 몇 개의 행에 극단적으로 크거나 작은 값을 할당하겠습니다.

mpg <- as.data.frame(ggplot2::mpg) # mpg 데이터 불러오기  
mpg[c(10, 14, 58, 93), "drv"] <- "k" # drv 이상치 할당  
mpg[c(29, 43, 129, 203), "cty"] <- c(3, 4, 39, 42) # cty 이상치 할당

이상치가 들어있는 mpg 데이터를 활용해서 문제를 해결해보세요.

구동방식별로 도시 연비가 다른지 알아보려고 합니다. 분석을 하려면 우선 두 변수에 이상치가 있는지 확인하려고 합니다.

* Q1. drv에 이상치가 있는지 확인하세요. 이상치를 결측 처리한 다음 이상치가 사라졌는지 확인하세요. 결측 처리 할 때는 %in% 기호를 활용하세요.
* Q2. 상자 그림을 이용해서 cty에 이상치가 있는지 확인하세요. 상자 그림의 통계치를 이용해 정상 범위를 벗어난 값을 결측 처리한 후 다시 상자 그림을 만들어 이상치가 사라졌는지 확인하세요.
* Q3. 두 변수의 이상치를 결측처리 했으니 이제 분석할 차례입니다. 이상치를 제외한 다음 drv별로 cty 평균이 어떻게 다른지 알아보세요. 하나의 dplyr 구문으로 만들어야 합니다.